

REMARKS

Applicant is in receipt of the Office Action mailed January 4, 2007. Claims 1-81 have been cancelled. New claims 82-102 have been added. Thus, claims 82-102 are pending in the case. Reconsideration of the present case is earnestly requested in light of the following remarks.

Objections

Claims 34-38 were objected to for use of the term “memory medium” without indicating that it is a memory medium of a computer. Applicant has replaced claims 1-81 with new claims 82-102, including claims 82-101, which refer instead to “a computer readable memory medium”. Applicant thus respectfully requests removal of the objection to these claims.

Section 101 Rejections

Claims 1-33 and 39-81 were rejected under 35 U.S.C. 101 for being directed to non-statutory subject matter, specifically, for claiming “a node”, without being supported by hardware such as tangible computer storage or execution engine. Applicant has replaced these claims with new claims that specify such a tangible computer storage or execution engine, and respectfully requests removal of the section 101 rejection of these claims.

Claims 39 and 81 were rejected under 35 U.S.C. 101 for being directed to “a memory medium” without being supported by hardware such as tangible computer storage or execution engine. Applicant has replaced these claims with new claims that specify such a tangible computer storage or execution engine, and respectfully requests removal of the section 101 rejection of these claims.

Section 102 Rejections

Claims 1-81 were rejected under 35 U.S.C. 102(e) as being anticipated by Sojoodi et al (US 6,437,805 B1, “Sojoodi”). Applicant has replaced original claims 1-81 with new claims 82-102, as indicated above, thus rendering rejection of the original claims

moot. However, Applicant has responded to the Examiner's arguments as applied to the new claims.

New claim 82 recites:

82. A computer readable memory medium comprising program instructions, wherein the program instructions are executable by a processor to:

display a function node in a graphical program on a display, wherein the graphical program comprises a plurality of nodes and connections between the plurality of nodes, wherein the plurality of connected nodes visually indicate functionality of the graphical program, and wherein the function node is executable in the graphical program to perform a first function;

display a function specific property node in the graphical program on the display, wherein the function specific property node is specific to the first function, wherein the function specific property node comprises a plurality of properties of the first function;

associate the function specific property node with the function node;

display the plurality of properties on the display; and

receive user input selecting one or more of the plurality of properties;

wherein the selected one or more properties are accessible during execution of the graphical program.

Nowhere does Sojoodi disclose program instructions executable to "display a function node in a graphical program on a display, wherein the graphical program comprises a plurality of nodes and connections between the plurality of nodes, wherein the plurality of connected nodes visually indicate functionality of the graphical program, and wherein the function node is executable in the graphical program to perform a first function; and to **display a function specific property node in the graphical program on the display, wherein the function specific property node is specific to the first function, wherein the function specific property node comprises a plurality of properties of the first function,**" as recited in claim 82.

Applicant respectfully submits that per Sojoodi's Abstract, Sojoodi is directed to a system and method for creating a graphical program that is operable to access *capabilities of an object*. In col.13:33-40, Sojoodi discloses various "object nodes" that are configured by a user to perform various functions related to the object, including, for example, an object function node and an object property node. Note that per Sojoodi's Specification, the object property node is operable to provide access to properties *of the object*, not of a function of a function node of a graphical program, as discussed in the following text:

Col.15:32-38:

In step 306 the user then configures the object node to receive information on the object, preferably by the user configuring the object node with a reference to the object, e.g., a pointer, address, or other information which specifies the identity and/or location of the object. (*emphasis added*)

Col.15:56-60

Step 306 also preferably includes the user selecting the class of the object. Once the class is selected, then the object is preferably instantiated at run time. Step 306 further includes selecting any desired methods to be invoked on the object or properties to get/set on the object. For example, if the object node is an invoke node, the user preferably selects one or more methods which are to be invoked on the object by the invoke node during execution of the graphical program. If the object node is a property node, the user preferably selects one or more properties to get/set on the object during execution of the graphical program. (*emphasis added*)

Clearly, Sojoodi's property node is for accessing *object* properties, not properties of a function of a function node. The Office Action itself characterizes Sojoodi's property node similarly, quoting col.17:55-65: "...perform functions or access capabilities of automation objects..." Moreover, Applicant respectfully notes that while Sojoodi discloses that the object may be an application, software component, or software object, e.g., per standard object-oriented software principles, nowhere does Sojoodi specifically teach that the object is a function node. In fact, Sojoodi specifically indicates otherwise. For example, as mentioned above, Sojoodi lists the various object nodes as including an object property node, and object function node, an object open node, and so

forth. Since there is already an object function node directed to the object, the object is clearly not itself a function node.

Thus, Applicant respectfully submits that Sojoodi nowhere discloses a function specific property node that is specific to a function of a function node, where the function specific property node includes a plurality of properties of the function (of the function node), and where the function-specific property node allows access to a user-selected one or more properties of the plurality of properties during execution of the graphical program.

Thus, for at least the reasons provided above, Applicant respectfully submits that Sojoodi fails to teach or suggest all the features and limitations of claim 82, and so claim 82 and those claims dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

Independent claim 102 includes similar limitations as claim 82, and so the above arguments apply with equal force to this claim. Thus, for at least the reasons provided above, Applicant respectfully submits that claim 102 and those claims dependent therefrom are patentably distinct and non-obvious over the cited art, and are thus allowable.

Applicant further submits that numerous ones of the dependent claims recite further distinctions over the cited art. For example, new claim 83, similar to originally filed claim 2, includes the limitation: **wherein the property node is statically typed to correspond to the function node.** Nowhere does Sojoodi disclose this limitation.

Applicant notes that the cited Figure 4:306, and related text, make no mention of such static typing of a property node. This text is quoted above, and, as may be seen, clearly explains that the user selects the class type of the object, then provides references to the object to the various object nodes. The object nodes (e.g., the object property node) then use this reference to access the object, e.g., to retrieve properties, methods, etc. Thus, Sojoodi's object nodes are functionally polymorphic, basing their behavior on the object type provided by the user during configuration. As those of skill in the programming arts readily understand, such polymorphic behavior is not possible with statically typed nodes. In other words, Sojoodi's property node is generic until the user configures the node with the object reference, at which time the property node behaves

according to the object type. In direct contrast, a property node that is statically typed to correspond to a particular function node is *not* able to change its functionality based on a user-configured function node type, but rather is constrained to the behavior corresponding to the function node type in accordance with the node's static typing. Thus, as an example, if the function node of claim 82 were a read node, executable to perform a read function, the function-specific property node would be a statically typed read property node, and could not be any other type of property node, i.e., cannot be polymorphic.

Thus, Applicant respectfully submits that claim Sojoodi fails to teach or suggest this limitation of claim 83.

Applicant also further asserts that numerous other ones of the dependent claims recite further distinctions over the cited art. However, since the independent claims have been shown to be patentably distinct, a further discussion of the dependent claims is not necessary at this time.

Applicant believes that the above arguments obviate a section 102 rejection of the new claims.

CONCLUSION

Applicant submits the application is in condition for allowance, and an early notice to that effect is requested.

If any extensions of time (under 37 C.F.R. § 1.136) are necessary to prevent the above-referenced application(s) from becoming abandoned, Applicant(s) hereby petition for such extensions. The Commissioner is hereby authorized to charge any fees which may be required or credit any overpayment to Meyertons, Hood, Kivlin, Kowert & Goetzel P.C., Deposit Account No. 50-1505/5150-81000/JCH.

Also filed herewith are the following items:

- Request for Continued Examination
- Terminal Disclaimer
- Power of Attorney By Assignee and Revocation of Previous Powers
- Notice of Change of Address
- Other:

Respectfully submitted,

/Jeffrey C. Hood/

Jeffrey C. Hood, Reg. #35198
ATTORNEY FOR APPLICANT(S)

Meyertons, Hood, Kivlin, Kowert & Goetzel PC
P.O. Box 398
Austin, TX 78767-0398
Phone: (512) 853-8800
Date: April 4, 2007 JCH/MSW